(19)日本国特許庁（ＪＰ）　(12) **公開特許公報（A）**　(11)特許出願公開番号

特開平10－209993

(43)公開日　平成10年(1998)８月７日

(51)Int.Cl.[6]	識別記号	ＦＩ	
Ｈ０４Ｊ　3/00		Ｈ０４Ｊ　3/00	Ｍ
Ｈ０４Ｎ　7/24		Ｈ０４Ｎ　7/13	Ｚ

審査請求　未請求　請求項の数８　ＯＬ　（全 12 頁）

(21)出願番号　特願平9－12808

(22)出願日　平成９年(1997)１月27日

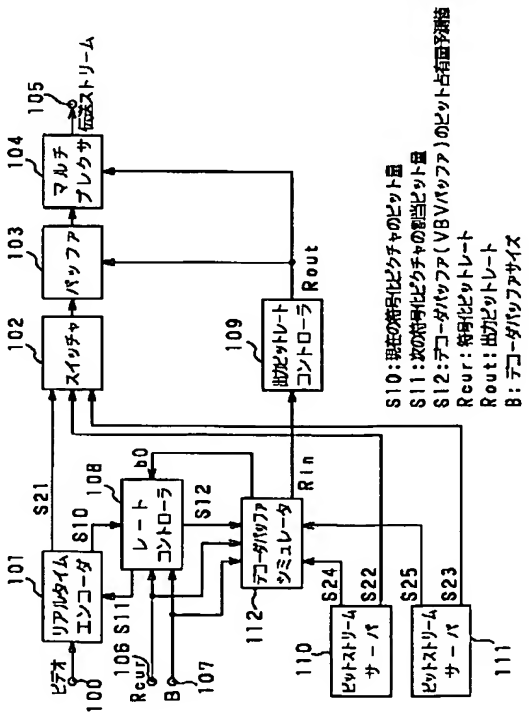
(71)出願人　000002185
ソニー株式会社
東京都品川区北品川６丁目７番35号
(72)発明者　加藤　元樹
東京都品川区北品川６丁目７番35号　ソニー株式会社内
(72)発明者　小柳　秀樹
東京都品川区北品川６丁目７番35号　ソニー株式会社内
(74)代理人　弁理士　小池　晃　（外２名）

(54)【発明の名称】　符号化信号伝送方法及び装置

(57)【要約】

【課題】　２つのビットストリームを継ぎ合わせて伝送する時に、デコーダ側でシームレスな画像再生を保証する。

【解決手段】　入力ビットストリームを一時的に蓄えるエンコーダバッファ１０３を設け、短時間の入力符号量の変動を平滑化し、エンコーダバッファ１０３から所定のビットレートでビットストリームを出力するシステムにおいて、２つのビットストリームＳＴ１，ＳＴ２を継ぎ合わせて伝送する時に、時間的に後ろとなるビットストリームをデコード開始する時に必要なデコーダバッファの初期ビット占有量に基づいて、エンコーダバッファ１０３からの出力ビットレートを制御するためのデコーダバッファシミュレータ１１２と出力ビットレートコントローラ１０９とを有する。



1

【特許請求の範囲】

【請求項１】　ディジタル信号を符号化した入力ビットストリームを一時的に蓄える送信バッファを設け、短時間の入力符号量の変動を平滑化し、当該送信バッファから所定のビットレートでビットストリームを出力する符号化信号伝送方法において、

２つのビットストリームを継ぎ合わせて伝送する時に、時間的に後ろとなるビットストリームをデコード開始する時に必要なデコーダバッファの初期ビット占有量に基づいて、上記送信バッファからの出力ビットレートを制御することを特徴とする符号化信号伝送方法。

【請求項２】　２つのビットストリームを継ぎ合わせて伝送する時に、時間的に後ろとなるビットストリームをデコード開始する時のデコーダバッファのビット占有量が、当該ビットストリームのデコードに必要な初期ビット占有量に等しいか又は当該初期ビット占有量以上になるように、上記送信バッファからの出力ビットレートを制御することを特徴とする請求項１記載の符号化信号伝送方法。

【請求項３】　ディジタル信号をリアルタイムにて符号化するときのビットストリームの最後でのデコーダバッファのビット占有量を予測し、

上記時間的に後ろとなるビットストリームをデコード開始する時に必要なデコーダバッファの初期ビット占有量と、上記予測値とに基づいて、上記送信バッファからの出力ビットレートを制御することを特徴とする請求項１記載の符号化信号伝送方法。

【請求項４】　上記ディジタル信号をリアルタイムにて符号化するときの符号化ビットレートと、上記ディジタル信号をリアルタイムにて符号化したときの所定単位毎の符号量と、上記デコーダバッファの初期ビット占有量と、上記デコーダバッファのサイズとから、次ぎに符号化する複数の所定単位のディジタル信号に対する割当符号量を計算し、上記符号量と上記割当符号量との誤差分が小さいときに、上記予測を行うことを特徴とする請求項３記載の符号化信号伝送方法。

【請求項５】　ディジタル信号を符号化した入力ビットストリームを一時的に蓄える送信バッファを設け、短時間の入力符号量の変動を平滑化し、当該送信バッファから所定のビットレートでビットストリームを出力する符号化信号伝送装置において、

２つのビットストリームを継ぎ合わせて伝送する時に、時間的に後ろとなるビットストリームをデコード開始する時に必要なデコーダバッファの初期ビット占有量に基づいて、上記送信バッファからの出力ビットレートを制御する送信バッファ出力ビットレート制御手段を有することを特徴とする符号化信号伝送装置

【請求項６】　２つのビットストリームを継ぎ合わせて伝送する時に、上記送信バッファ出力ビットレート制御手段は、時間的に後ろとなるビットストリームをデコード開始する時のデコーダバッファのビット占有量が、当該ビットストリームのデコードに必要な初期ビット占有量に等しいか又は当該初期ビット占有量以上になるように、上記送信バッファからの出力ビットレートを制御することを特徴とする請求項５記載の符号化信号伝送装置。

2

【請求項７】　ディジタル信号をリアルタイムにて符号化するときのビットストリームの最後でのデコーダバッファのビット占有量を予測する予測手段を設け、

上記送信バッファ出力ビットレート制御手段では、上記時間的に後ろとなるビットストリームをデコード開始する時に必要なデコーダバッファの初期ビット占有量と、上記予測手段からの予測値とに基づいて、上記送信バッファからの出力ビットレートを制御することを特徴とする請求項５記載の符号化信号伝送装置。

【請求項８】　上記予測手段は、上記ディジタル信号をリアルタイムにて符号化するときの号化ビットレートと、上記ディジタル信号をリアルタイムにて符号化したときの所定単位毎の符号量と、上記デコーダバッファの初期ビット占有量と、上記デコーダバッファのサイズとから、次ぎに符号化する複数の所定単位のディジタル信号に対する割当符号量を計算し、上記符号量と上記割当符号量との誤差分が小さいときに、上記予測を行うことを特徴とする請求項７記載の符号化信号伝送装置。

【発明の詳細な説明】

【０００１】

【産業上の利用分野】本発明は、送信側にてディジタル信号を可変ビットレートで符号化し、これを受信側へ可変ビットレートで伝送する場合に用いて好適な符号化信号伝送方法及び装置に関する。

【０００２】

【従来の技術】テレビジョン放送などにおいて、２つのビデオ素材を継ぎ合わせて伝送する場合、受信側にて継ぎ目がわからないようにシームレスに再生できることが重要である。アナログビデオ素材や非圧縮のディジタルビデオ素材において、２つのビデオ素材がシームレスに再生されるように、継ぎ合わすスイッチング操作は、放送スタジオなどでよく用いられている。

【０００３】近年は、ディジタルビデオ信号の高能率符号化技術の進歩により、ディジタルビデオ信号を限られた伝送路で効率良く伝送することが可能となった。ここで、２つの圧縮符号化ビデオビットストリームを継ぎ合わせて、シームレスに再生されるように伝送する技術は、従来のアナログビデオ素材や非圧縮のディジタルビデオ素材を継ぎ合わせて伝送する場合に比べて複雑である。この技術を複雑にする大きな理由は、受信側に設けられている後述する受信バッファのビット占有量の管理が必要なためである。このバッファ管理について以下に簡単に説明する。

【０００４】図７には、送信側におけるエンコーダへの

ビデオ信号の入力から、受信側におけるデコーダからのビデオ信号の出力までのシステム構成を示す。

【０００５】受信側（以下エンコーダ側という）のビデオエンコーダ１０では、複数のピクチャからなる入力ビデオ信号をエンコードする。このビデオエンコーダ１０にて発生する各入力ピクチャの符号化ビット量は、これら各ピクチャ毎に変化するものである。このため、この図７に示すシステムのエンコーダ側では、上記ビデオエンコーダ１０の符号化出力を一時的に送信バッファ（以下、エンコーダバッファと呼ぶ）２０に蓄え、それを決められた出力ビットレートのビットストリームとして読み出すようにしている。このエンコーダバッファ２０から読み出されたビットストリームは、多重化器（マルチプレクサ）３０へ入力され、この多重化器３０では他の音声データ等のビットストリームと上記ビデオのビットストリームとを多重化して出力する。この出力されたビットストリームは、伝送路を介して受信側（以下デコーダ側という）に伝送される。そして、デコーダ側の分離器（デマルチプレクサ）４０では、上記エンコーダ側から伝送されたビットストリームの中からユーザにて指定されたビットストリームが分離されて、それが受信バッファ（以下デコーダバッファと呼ぶ）５０へ入力される。デコーダバッファ５０は、後段のビデオデコーダ６０がピクチャ毎に読み出すビット量の変動を吸収するために設けられている。ビデオデコーダ６０は、デコーダバッファ５０からビットストリームを読みだし、それをデコードして、再生動画像信号を生成して出力する。

【０００６】ここで、上述のような動画像符号化方法の代表的なものにいわゆるＭＰＥＧがある。ＭＰＥＧはデコーダバッファ５０の理想的な入出力モデルを、ＶＢＶ(Video Buffering Verifier)モデルとして規定しており、それはＩＳＯ／ＩＥＣ　１１１７２－２　Ａｎｎｅｘ　ＣまたはＩＳＯ／ＩＥＣ　１３８１－２　Ａｎｎｅｘ　Ｃに記述されている。ＶＢＶモデルのバッファをＶＢＶバッファという。ＶＢＶバッファサイズは、ＭＰＥＧビットストリームの中の "vbv_buffer_size"で指示される。その標準的な大きさは、例えば、ＭＰＥＧ２のＭＰ@ＭＬ(Main Prifile at Main Level)では、１．７５Ｍｂｉｔである。

【０００７】デコーダのＶＢＶは、以下の理想条件のもとで動作することを仮定している。

【０００８】(1)エンコーダとＶＢＶは、同じピクチャレートで同期して動作する。

【０００９】(2)各ピクチャに対するビットストリームはデコーダバッファから瞬時に取り除かれ、各ピクチャのデコードは瞬時に行なわれる。

【００１０】したがってエンコーダは、ＶＢＶバッファモデルを想定しながら、ＶＢＶバッファをオーバフロウとアンダフロウをさせないように注意してエンコードしなければならない。

【００１１】ところでＶＢＶを満たすように符号化された２つの独立したＭＰＥＧ２ビットストリームを単純に継ぎ合わせて伝送する場合、一般にデコーダ側でのシームレスなビデオ再生は保証されない。その場合、発生する問題としては、デコーダバッファのオーバフロウによる再生画像の乱れ、またはデコーダバッファのアンダフロウによる再生画像のフリーズ、などの問題が起こる。

【００１２】図８は、第１，第２の２つのビットストリームを継ぎ合わせた結果、デコーダバッファがアンダフロウする例を説明する図である。図８の(a)はＶＢＶを満たすように符号化された第１のストリームＳＴ１を再生する時のデコーダバッファのビット占有量の軌跡を示している。横軸ｔは時間の経過を表す。また縦軸は、デコーダバッファへ入力されるビットストリームのある時刻までのビット量の累計値を表す。直線ａ－ｂの傾きは、デコーダバッファへの入力ビットレートＲ１を表す。直線ｃ－ｄと直線ａ－ｂの縦軸方向の幅は、デコーダバッファ（ＶＢＶバッファ）サイズＢを表し、当該バッファサイズＢは一定である。Ａ(i)は、ｉ（ｉ＝０，１，２，・・・）番目の符号化ピクチャを表し、その大きさはその符号化ピクチャのビット量を表す。Ｄ(i)は、ｉ番目の符号化ピクチャＡ(i)をデコードする時刻を表す。デコードするピクチャの間隔（すなわちＤ(i+1)-Ｄ(i)）は、例えば、いわゆるＮＴＳＣビデオ信号では１／２９．９７秒であり、いわゆるＰＡＬビデオ信号では１／２５秒となる。デコーダバッファのビット占有量の変化は、図中ジグザグの階段状の軌跡と直線ａ－ｂの縦軸方向の幅により示される。階段状の軌跡の縦軸方向の動きは、デコーダバッファから瞬時に符号化ピクチャが読み出されて取り除かれたことを表し、また階段状の軌跡の横軸方向の動きは、デコーダバッファからのビットストリームの読み出しは停止していて、デコーダバッファへビットレートＲ１でビットストリームが入力されていることを表す。

【００１３】デコーダバッファのビット占有量の変化を以下に説明する。ｔ＝０からデコーダバッファへビットレートＲ１でビットストリームの入力が始まる。そして、ｔ＝ｓｐ１で０番目の符号化ピクチャＡ(0)のデコードをする。このデコード開始時刻は、ＭＰＥＧビットストリーム中の"vbv_delay"または"decoding_time_stamp"で指示されるものであり、第１のストリームＳＴ１をデコードするために必要なスタートアップの遅延時間である。

【００１４】デコーダバッファのビット占有量は、ｔ＝Ｄ(0)（ｔ＝ｓｐ１と同じ）における０番目のピクチャＡ(0)のデコードによって瞬時に当該０番目のピクチャＡ(0)分だけビット量が減少する。続いて、ｔ＝Ｄ(1)になるまで、デコーダバッファへはビットレートＲ１で１番目のピクチャＡ(1)のビットストリームが入力されるため、デコーダバッファのビット占有量は時間とともに

増加する。次ぎにｔ＝Ｄ(1)では、当該１番目のピクチャＡ(1)のデコードによってデコーダバッファからは、瞬時に当該１番目のピクチャＡ(1)分のビット量だけビット占有量が減少する。以下同様に、定められたデコード時間間隔で各ピクチャのデコードを継続する。

【００１５】図８の(b)は、図８の(a)と同様にして、ＶＢＶを満たすように符号化された第２のストリームＳＴ２を再生する時のデコーダバッファのビット占有量の軌跡をしている。ここではデコーダバッファへの入力ビットレートはＲ２とする。ここで、第２のストリームＳＴ２をｎ番目の符号化ピクチャＡ(n)からデコード開始するために必要な初期ビット占有量は図中縦軸方向の幅Ｏｓである。この図８の(b) の第２のストリームＳＴ２におけるデコーダバッファのビット占有量の変化についての説明は図８の(a)と同様に考えることができるため省略する。

【００１６】図８の(c)は、上述した第１のストリームＳＴ１と第２のストリームＳＴ２を、図８の(a)の第１のストリームＳＴ１のｔ＝ｅｐ１と図８の(b)の第２のストリームＳＴ２のｔ＝ｓｐ２とで単純に継ぎ合わせたビットストリームをデコードした時のデコーダバッファのビット占有量の軌跡を示している。この図８の(c)からは、第２のストリームＳＴ２をデコードする時の初期ビット占有量が不足しているために、未来においてバッファアンダフロウが発生していることがわかる。

【００１７】つぎに、図９を用いて、２つのビットストリームを継ぎ合わせた結果、デコーダバッファがオーバフロウする例を説明する。なおこの図９では、図中の指示符号を図８と同様のものを使用している。

【００１８】図９の(a)は第１のストリームＳＴ１を再生する時のデコーダバッファのビット占有量の軌跡を示している。また、図９の(b)は、第２のストリームＳＴ２を再生する時のデコーダバッファのビット占有量の軌跡を示している。ここで、第２のストリームＳＴ２をｎ番目の符号化ピクチャA(n)からデコード開始するために必要な初期ビット占有量は図中の縦軸方向の幅Ｏｓである。

【００１９】図９の(c)は、上記第１のストリームＳＴ１と第２のストリームＳＴ２を、図９の(a)の第１のストリームＳＴ１のｔ＝ｅｐ１と図９の(b)の第２のストリームＳＴ２のｔ＝ｓｐ２とで単純に継ぎあわされたビットストリームをデコードした時のデコーダバッファのビット占有量の軌跡を示している。この図８の(c)からは、第２のストリームＳＴ２をデコード開始する時の初期ビット占有量が過剰であるために、未来においてバッファオーバフロウが発生していることがわかる。

【００２０】このように、２つの符号化ビットストリームを単純に継ぎ合わせて伝送する場合、デコーダ側でのシームレスなビデオ再生は必ずしも保証されない。

【００２１】２つのビットストリームを継ぎ合わせて伝送する時に、上述のような問題を発生させないで、デコーダ側でシームレスに再生ができるようにするためには、時間的に後ろとなるビットストリームをデコード開始する時のデコーダバッファのビット占有量が、そのストリームのデコードに必要な初期ビット占有量だけあれば良い。すなわち例えば、図８の例での第２のストリームＳＴ２のデコード開始時の初期ビット占有量Ｏｓが、第１のストリームＳＴ１の最後のビット占有量Ｏｅに等しくなるように、第２のストリームＳＴ２がエンコードされていれば良い。または、第１のストリームＳＴ１の上記最後のビット占有量Ｏｅが、第２のストリームＳＴ２の初期ビット占有量Ｏｓに等しくなるように、第１のストリームＳＴ１がエンコードされていれば良い。

【００２２】

【発明が解決しようとする課題】上述したように、２つのビットストリームを継ぎ合わせて伝送する時に、デコーダ側でシームレスに画像再生ができるためには、時間的に後ろとなるビットストリームをデコード開始する時のデコーダバッファのビット占有量が、そのストリームのデコードに必要な初期ビット占有量だけあることが保証されなければならない。

【００２３】この条件が満たされていない場合は、それを満たすように、２つの継ぎ合わせるビットストリームの片方または両方を再エンコードすれば良い。しかし、再エンコードは、処理時間が多くなることや画質劣化が問題になるため良い方法ではない。

【００２４】このような背景からわかるように、従来は２つのビットストリームを継ぎあわせて伝送する時に、デコーダ側でシームレスに再生ができることを保証するための方法が不十分であった。

【００２５】そこで、本発明は、このような実情に鑑みてなされたものであり、２つのビットストリームを継ぎ合わせて伝送する時に、デコーダ側でシームレスな画像再生を保証する符号化信号伝送方法及び装置を提供することを目的とするものである。

【００２６】なお、この時、継ぎ合わせる２つのビットストリームは、既に符号化されているものでも良いが、リアルタイムにエンコードしてビットストリームを継ぎ合わせた場合には、そのエンコード条件に制限を加えなくても良いものであることが望ましい。

【００２７】

【課題を解決するための手段】本発明は、上述した課題を解決するために、ディジタル信号を符号化したビットストリームを伝送するための符号化信号伝送方法及び装置であり、入力ビットストリームを一時的に蓄える送信バッファを設け、短時間の入力符号量の変動を平滑化し、送信バッファから所定のビットレートでビットストリームを出力するシステムにおいて、２つのビットストリームを継ぎ合わせて伝送する時に、時間的に後ろとなるビットストリームをデコード開始する時に必要なデコ

ーダ（受信）バッファの初期ビット占有量に基づいて、上記送信バッファからの出力ビットレートを制御することを特徴としている。

【００２８】具体的には、２つのビットストリームを継ぎ合わせて伝送する時に、時間的に後ろとなるビットストリームをデコード開始する時のデコーダ（受信）バッファのビット占有量が、そのストリームのデコードに必要な初期ビット占有量に等しい、又はそれ以上になるように、上記送信バッファからの出力ビットレートを制御することを特徴とする。

【００２９】

【発明の実施の形態】本発明に係る符号化信号伝送方法を実現する符号化信号伝送装置の実施の形態について、図面を参照しながら説明する。

【００３０】図１には、本発明の符号化信号伝送方法を適用した符号化信号伝送装置の構成例を示す。

【００３１】始めに、既に符号化されている２つのビットストリームＳ２２とＳ２３を継ぎ合わせて伝送する場合について説明する。

【００３２】この場合、ビットストリームＳ２２とＳ２３が、それぞれビットストリームサーバ１１０と１１１から読み出され、スイッチャ１０２へ供給される。スイッチャ１０２は、伝送するビデオプログラムのスケジュールに従って、ビットストリームＳ２２とＳ２３をスイッチングして、バッファ（エンコーダバッファ）１０３へ供給する。バッファ１０３は出力ビットレートコントローラ１０９から指示された出力ビットレートRoutに従ってビットストリームを出力する。マルチプレクサ１０４は、出力ビットレートRoutに従ってエンコーダバッファ１０３からビットストリームを読みださせ、それを他の入力ストリーム（図示していないが例えば音声のビットストリーム等）と多重化して、端子１０５から伝送ストリームとして出力する。出力ビットレートRoutは、デコーダバッファシミュレータ１１２が計算するところの受信側のデコーダバッファの入力ビットレートRinのスケジュールに基づいて計算される。

【００３３】デコーダバッファシミュレータ１１２へは、ビットストリームＳ２２とＳ２３のそれぞれについての情報信号Ｓ２４とＳ２５が入力される。上記情報信号Ｓ２４は、ビットストリームＳ２２の符号化ビットレート，デコーダバッファサイズ，各符号化ピクチャの大きさやそのピクチャからデコード開始する時に必要なデコードバッファのビット占有量（初期ビット占有量を含む）などからなる。情報信号Ｓ２５は、ビットストリームＳ２３についての同様の情報である。

【００３４】図２と図３を例として、出力ビットレートRoutの制御方法を説明する。なお、図２と図３の各指示符号は、前記図８に準じたものを使用している。本実施の形態では、図２の(a)の第１のストリームＳＴ１を上記ビットストリームＳ２２とし、図２の(b)の第２のストリームＳＴ２を上記ビットストリームＳ２３とする。

【００３５】図２の(a)，(b)は、それぞれ従来の技術で説明した図８の(a)，(b)と同じ図であり、ＶＢＶを満たすように符号化された第１のストリームＳＴ１，第２のストリームＳＴ２を再生する時のデコーダバッファのビット占有量の軌跡を示している。ここで、第１のストリームＳＴ１は符号化ビットレートＲ１（デコーダバッファへの入力ビットレートがＲ１、すなわちエンコーダバッファの出力ビットレートがＲ１）でエンコードされており、第２のストリームＳＴ２は符号化ビットレートＲ２（デコーダバッファへの入力ビットレートがＲ２、すなわちエンコーダバッファの出力ビットレートがＲ２）でエンコードされている。第１のストリームＳＴ１について、デコーダバッファへの入力ビットレートがＲ１の時（すなわちエンコーダバッファの出力ビットレートがＲ１の時）、ｔ＝ｅｐ１でのデコーダバッファの最後のビット占有量は図中縦軸方向の幅Ｏｅである。また、第２のストリームＳＴ２について、デコーダバッファへの入力ビットレートがＲ２の時（すなわちエンコーダバッファの出力ビットレートがＲ２の時）、ｔ＝ｓｐ２でのデコーダバッファの初期ビット占有量は図中縦軸方向の幅Ｏｓである。

【００３６】図２の(c)は、２つのビットストリームの第１のストリームＳＴ１と第２のストリームＳＴ２を継ぎ合わせて伝送する時に、前記図８で示したようなバッファアンダフロウの問題が起こらないようにデコードバッファへの入力ビットレートをコントロールする時、言い換えればエンコーダバッファの出力ビットレートをコントロールした時のデコーダバッファのビット占有量の軌跡を示している。ここでは、第２のストリームＳＴ２のｔ＝ｓｐ２におけるバッファの初期占有量がＯｓとなるように、折れ線ａ－ｋ－ｅ－ｆで示すように、エンコーダバッファの出力ビットレートが制限、すなわちデコーダバッファへの入力ビットレートが制御されている。具体的に言うと、ｔ＝αからデコーダバッファへの入力ビットレートを変化させて、ｔ＝ｓｐ２（ｔ＝ｅｐ１と同一）におけるデコーダバッファの初期ビット占有量がＯｓになるように、エンコーダバッファの出力ビットテレビジョンを制御するようにしているので、第２のストリームＳＴ２の再生においてデコーダバッファが破綻することはない。すなわち、このようにすることにより、前記図８の(a)，(b)のような第１のストリームＳＴ１と第２のストリームＳＴ２を継ぎ合わせたとしてもデコーダバッファがアンダフロウすることはなく、従って第１のストリームＳＴ１と第２のストリームＳＴ２をシームレスに画像再生することができる。

【００３７】ここで、入力ビットレートRin(t)でデコーダバッファへビットストリームが入力される時のデコーダバッファのビット占有量は、(1-1)式に基づいて計

算される。

【００３８】

$$b0 = R1 \times sp1$$

【数１】

$$Od(k) = b0 + \sum_{i=0}^{k-1}\{(D(i+1) - D(i)) \times Rin(i)\} - \sum_{i=0}^{k-1} A(i) \qquad (1\text{-}1)式$$

$$0 \leqq Od(k) \leqq B$$

【００３９】ここで、

Ｏｄ(ｉ)：ｉ番目のピクチャをデコードする直前でのデコーダバッファのビット占有量。

【００４０】ｂ０　　：デコード開始時のデコーダバッファの初期ビット占有量（＝Ｏｓ）。

【００４１】Ｄ(ｉ)　：ｉ番目のピクチャのデコード時刻。

【００４２】Ａ(ｉ)　：ｉ番目のピクチャの符号化ビット量。

【００４３】Ｒｉｎ(ｉ)：ｉ番目のピクチャＡ(ｉ)とｉ＋１番目のピクチャＡ(ｉ+1)の間隔でのデコーダバッファへの入力ビットレート。

【００４４】Ｂ　　：デコーダバッファサイズ。

【００４５】各ピクチャＡ(ｉ)のサイズは、情報信号Ｓ２４，Ｓ２５から指示される。ピクチャＡ(0)がデコード開始される時刻は、ｔ＝ｓｐ１である。この時刻はピクチャＡ(0)からデコード開始する時に必要なデコードバッファの初期ビット占有量ｂ０（＝Ｏｓ）に基づいている。

【００４６】デコーダバッファシミュレータ１１２は、(1-1)式に基づいて、第１のストリームＳＴ１の最後ｔ＝ｅｐ１のビット占有量が上記初期ビット占有量Ｏｓになるように、ｔ＝αから入力ビットレートＲｉｎが変更されるようにするスケジュールを立てる。すなわち、デコーダバッファシミュレータ１１２は、図２の(c)の折れ線ａ－ｋ－ｅ－ｆで示されるようなデコーダバッファの入力ビットレートＲｉｎ(t)のスケジュールを立てる。

【００４７】このデコーダバッファシミュレータ１１２からの上記シミュレートにより得られた入力ビットレートＲｉｎが上記出力ビットレートコントローラ１０９に送られ、この出力ビットレートコントローラ１０９が上記入力ビットレートＲｉｎに基づいてエンコーダバッファ１０３の出力ビットレートＲｏｕｔを制御する。これにより、デコーダバッファ１０３の出力ビットレートＲｏｕｔは、上記シミュレートにより得られたデコーダバッファの入力ビットレートＲｉｎに同期して変化することになる。このことについて図３を用いて説明する。

【００４８】図３は、図２の(c)に示したようなデコードバッファの入力ビットレート制御すなわちエンコーダバッファの出力ビットレートを制御した時のエンコーダ側とデコーダ側のバッファのビット占有量の変化の例を示す。図１のエンコーダバッファ１０３は前記図７のエンコーダバッファ２０と同じ働きを有するものであり、以下に説明するデコーダバッファは図７のデコーダバッファ５０に対応するものである。

【００４９】図３において、折れ線ａ－ｋ－ｅ－ｆから右側は、デコーダ側のバッファ（デコーダバッファ）のビット占有量の変化を表し、同線から左側がエンコーダバッファのビット占有量の変化を表す。横軸ｔは時間の経過を表す。ここで時間軸は２本引かれており、上の時間軸はエンコーダ側の時間の経過を表し、下の時間軸はデコーダ側の時間の経過を表す。簡単化のために折れ線ａ－ｋ－ｅ－ｆをエンコーダ側とデコーダ側で共有し、その間に時間差がないように書かれているが、実際には、エンコーダ側とデコーダ側の間には、一定の伝送路遅延時間Ｄ０が存在する。折れ線ａ－ｋ－ｅ－ｆの時刻ｔでの傾きは、エンコーダ側から見ると、時刻ｔでのエンコーダバッファからの出力ビットレートＲｏｕｔを表し、デコーダ側から見ると、時刻ｔでのデコーダバッファへの入力ビットレートＲｉｎを表す。また縦軸は、エンコーダ側ではエンコーダバッファから出力されるビットストリームのある時刻までのビット量の累計値を表し、デコーダ側ではデコーダバッファへ入力されるビットストリームのある時刻までのビット量の累計値を表す。

【００５０】デコーダバッファのビット占有量の変化については、従来の技術のところで説明したので、ここではエンコーダバッファのビット占有量の変化についてだけ説明する。

【００５１】図中Ｂは、デコーダバッファ（ＶＢＶバッファ）サイズに等しい大きさである。また、Ａ(ｉ)は、ｉ番目の符号化ピクチャを表し、その大きさはその符号化ピクチャのビット量を表す。Ｅ(ｉ)は、ｉ番目のピクチャＡ(ｉ)がエンコーダバッファへ入力される時刻であり、ピクチャ間隔（すなわちＥ(ｉ+1)－Ｅ(ｉ)）は、デコードされるピクチャの間隔（すなわちＤ(ｉ+1)－Ｄ(ｉ)）に等しい。

【００５２】エンコーダバッファのビット占有量の変化は、図３中のエンコーダ側のジグザグの階段状の軌跡と折れ線ａ－ｋ－ｅ－ｆの縦軸方向の幅により示される。上記階段状の軌跡の縦軸方向の動きは、エンコーダバッファへ瞬時に符号化ピクチャが入力されたことを表し、また階段状の軌跡の横軸方向の動きは、エンコーダバッ

ファへのビットストリームの入力は停止していて、エンコーダバッファから折れ線ａ－ｋ－ｅ－ｆの傾きで示すビットレートでビットストリームが出力されていることを表す。すなわち、ｔ＝Ｅ(0)に第１のストリームＳＴ１のピクチャＡ(0)が瞬時にエンコーダバッファへ入力され、ビット占有量がこのピクチャＡ(0)の分だけ増加する。続いて、ｔ＝Ｅ(1)になるまで、エンコーダバッファからビットストリームが出力されるため、エンコーダバッファのビット占有量は時間とともに減少する。またｔ＝Ｅ(1)ではピクチャＡ(1)が瞬時にエンコーダバッファへ入力され、ビット占有量がこのピクチャＡ(1)の分だけ増加する。以下同様に、定められたピクチャ間隔

$$Rout(t-D0) = Rin(t)$$

ｔは時間

より具体的に図１を用いて説明すると、第１のストリームＳＴ１であるビットストリームＳ２２は、図３のｔ＝Ｅ(0)にエンコーダバッファ１０３へ入力される。そしてｔ＝（－Ｄ０）のときに前記に出力ビットレートＲ１でエンコーダバッファ１０３から第１のストリームＳＴ１として出力開始される。このとき、エンコーダバッファ１０３からの出力ビットレートRout(t)は、ｔ＝（－Ｄ０）からデコーダバッファシミュレータ１０９が立てた入力ビットレートRin(t)のスケジュールに基づいて制御されることになる。

【００５５】またもう一つの例として、図４の(a),(b)は、それぞれ従来の技術で説明した図９の(a),(b)と同じ図であり、ＶＢＶを満たすように符号化された第１のストリームＳＴ１，第２のストリームＳＴ２を再生する時のデコーダバッファのビット占有量の軌跡を示している。ここで、第１のストリームＳＴ１は符号化ビットレートＲ１（デコーダバッファへの入力ビットレートがＲ１、すなわちエンコーダバッファの出力ビットレートがＲ１）でエンコードされており、第２のストリームＳＴ２は符号化ビットレートＲ２（デコーダバッファへの入力ビットレートがＲ２、すなわちエンコーダバッファの出力ビットレートがＲ２）でエンコードされている。第１のストリームＳＴ１について、デコーダバッファへの入力ビットレートがＲ１の時（エンコーダバッファの出力ビットレートがＲ１の時）、ｔ＝ｅｐ１でのデコーダバッファの最後のビット占有量はＯｅである。また、第２のストリームＳＴ２について、デコーダバッファへの入力ビットレートがＲ２の時（エンコーダバッファの出力ビットレートがＲ２の時）、ｔ＝ｓｐ２でのデコーダバッファの初期ビット占有量はＯｓである。

【００５６】図４の(c)は、２つのビットストリームの第１のストリームＳＴ１と第２のストリームＳＴ２を継ぎ合わせて伝送する時に、図９で示したようなバッファオーバフロウの問題が起こらないように、デコードバッファへの入力ビットレートをコントロールする時、言い換えればエンコーダバッファの出力ビットレートをコントロールした時のデコーダバッファのビット占有量の軌跡を示している。ここでは、第２のストリームＳＴ２のｔ＝ｓｐ２におけるデコーダバッファの初期占有量がＯｓとなるように、折れ線ａ－ｋ－ｅ－ｆで示すように、エンコーダバッファの出力ビットレートが制御、すなわちデコーダバッファへの入力ビットレートが制御されている。具体的に言うと、ｔ＝αからデコーダバッファへの入力ビットレートを変化させて、ｔ＝ｓｐ２（ｔ＝ｅｐ１と同一）におけるデコーダバッファの初期ビット占有量がＯｓになるように、エンコーダバッファの出力ビットレートを制御するようにしているので、第２のストリームＳＴ２の再生においてデコーダバッファが破綻することはない。すなわち、このようにすることにより、前記図９の(a),(b)のような第１のストリームＳＴ１と第２のストリームＳＴ２を継ぎ合わせたとしてもデコーダバッファがオーバフロウすることはなく、従って第１のストリームＳＴ１と第２のストリームＳＴ２をシームレスに再生することが可能になる。

【００５７】より具体的に前記図１を用いて説明すると、デコーダバッファシミュレータ１１２は、前記(1-1)式に基づいて、第１のストリームＳＴ１の最後ｔ＝ｅｐ１のビット占有量が前記初期ビット占有量Ｏｓになるように、ｔ＝αから入力ビットレートが変更されるようにするスケジュールを立てる。すなわち、デコーダバッファシミュレータ１１２は、図４の(c)の折れ線ａ－ｋ－ｅ－ｆで示されるようなデコーダバッファへの入力ビットレートRin(t)のスケジュールを立てる。

【００５８】このデコーダバッファシミュレータ１１２からの上記シミュレートにより得られた入力ビットレートRinが上記出力ビットレートコントローラ１０９に送られ、この出力ビットレートコントローラ１０９が上記入力ビットレートRinに基づいて上記エンコーダバッファ１０３の出力ビットレートRoutを制御する。これにより、エンコーダバッファ１０３からの出力ビットレートRoutは、上記シミュレートにより得られたデコーダバッファへの入力ビットレートRinに同期して変化することになる。

【００５３】図２の(a),(b)の２つのビットストリームを継ぎ合わせて伝送する時に、図３に示すようにエンコーダバッファからの出力ビットレートを折れ線ａ－ｋ－ｅ－ｆで示されるように制御することにより、デコーダ側でのシームレスな再生を保証できる。すなわち図３の折れ線ａ－ｋ－ｅ－ｆは、エンコーダバッファからの出力ビットレートRoutとデコーダバッファへの入力ビットレートRinを示しており、送受信間の伝送路遅延時間Ｄ０を介して、RoutとRinは(1-2)式の関係がある。

【００５４】

(1-2)式

【００５９】図５は図４の(c)に示したようなデコードバッファへの入力ビットレート制御、すなわちエンコーダバッファの出力ビットレートを制御した時のエンコーダとデコーダのバッファのビット占有量の変化の例を示す。

【００６０】この図５の例においても、前記図３で説明したものと同様に、図４の(a),(b)の２つのビットストリームを継ぎ合わせて伝送する時に、図５に示すようにエンコーダバッファからの出力ビットレートRoutを折れ線ａ－ｋ－ｅ－ｆで示されるように制御することにより、デコーダ側でのシームレスな再生を保証できる。

【００６１】なお、デコーダバッファが、オーバフロウする前にデコーダバッファへのビットストリーム入力を停止できるようなアプリケーションの場合は、バッファオーバフロウは起こり得ない。このようなアプリケーションでは、第２のストリームＳＴ２をデコード開始する時のデコーダバッファのビット占有量は、そのストリームのデコードに必要な初期ビット占有量以上あってもよい。したがって、その場合は図９の(c)の折れ線ａ－ｂ－ｅ－ｆで示される入力ビットレートであっても、バッファオーバフロウは起こり得ない。

【００６２】次に、リアルタイムにエンコードするビットストリームと、既に符号化されたビットストリーム（以下の説明ではビットストリームＳ２２を用いた場合を例に挙げる）とを継ぎ合わせて伝送する場合について説明する。

【００６３】この場合は、リアルタイムエンコーダ１０１に対して端子１００がらビデオ信号が入力されて、このリアルタイムエンコーダ１０１にてリアルタイムに符号化が行われる。当該リアルタイムエンコーダ１０１から出力されるビットストリームＳ２１と、ビットストリームサーバ１１０から出力されるビットストリームＳ２２とは、スイッチャ１０２へ供給される。スイッチャ１０２は、伝送するビデオプログラムのスケジュールに従って、ビットストリームＳ２１とＳ２２をスイッチングして、エンコーダバッファ１０３へ供給する。エンコーダバッファ１０３は指示された出力ビットレートRoutに従ってビットストリームを出力する。マルチプレクサ１０４以降は前述同様である。

【００６４】また、レートコントーラ１０８には、端子１００を介してリアルタイムエンコーダ１０１に入力されるピクチャの符号化ビットレートRcurが、端子１０６から指示され、また端子１０７からデコーダバッファサイズBが指示される。

【００６５】レートコントーラ１０８は、上記符号化ビットレートRcurと、デコーダバッファサイズBと、リアルタイムエンコーダ１０１にてエンコードされた符号化ピクチャのビット量Ｓ１０と、デコーダバッファの初期ビット占有量ｂ０とを入力として、サイズBのデコーダバッファをオーバフロウやアンダフロウさせないように、端子１００から次に入力されて符号化するピクチャの割当ビット量Ｓ１１を計算し、リアルタイムエンコーダ１０１へ指定する。このときのレートコントローラ１０８は、(2-1)式, (2-2)式の演算を行う。

【００６６】

【数２】

$$k = 0 \text{ の時、}$$

$$Od(0) = b0$$

$$k \geqq 1 \text{ の時、}$$

$$Od(k) = b0 + \sum_{i=0}^{k-1}((D(i+1) - D(i)) \times Rcur) - \sum_{i=0}^{k-1} A(i) \quad \text{(2-1)式}$$

$$Od(k) + Rcur \times (D(k+1) - D(k)) - B \leqq AA(k) \leqq Od(k) \quad \text{(2-2)式}$$

$$\text{ただし、} AA(k) \geqq 0$$

【００６７】ここで、

Ｏｄ(i)　：ｉ番目のピクチャをデコードする直前でのデコーダバッファのビット占有量。

【００６８】ｂ０　　：デコード開始時のデコーダバッファの初期ビット占有量。

【００６９】Ｄ(i)　：ｉ番目のピクチャのデコード時刻。

【００７０】Ａ(i)　：ｉ番目のピクチャの符号化ビット量。

【００７１】Rcur　：ｉ番目のピクチャＡ(i)とｉ＋１番目のピクチャＡ(i+1)の間隔での符号化ビットレート(デコーダバッファへの入力ビットレート)。

【００７２】Ｂ　　：デコーダバッファサイズ。

【００７３】図１では、ｉ番目のピクチャA(i)のビット量がＳ１０であり、ＡＡ(i)が割当ビット量Ｓ１１となる。レートコントローラ１０８は、上記割当ビット量Ｓ１１として数フレーム未来までのフレームに対する割当ビット量を決定しており、ビット量Ｓ１０と割当ビット量Ｓ１１の誤差が小さい場合には、数フレーム未来の時点でのデコーダバッファのビット占有量Ｏｄをあらかじめ予測できる。

【００７４】これにより、レートコントーラ１０８は、

リアルタイムエンコーダ１０１で符号化されるビットストリームＳ２１の最後でのデコーダバッファのビット占有量Ｏｄを予測して、その値をＳ１２としてデコーダバッファシミュレータ１１２へ入力する。例えば、ビットストリームＳ２１が図２の(a)の第１のストリームＳＴ１である時、上記予測値Ｓ１２は前記最後のビット占有量Ｏｅとなる。

【００７５】また、このときのデコーダバッファシミュレータ１１２へは、ビットストリームサーバ１１０から出力されるビットストリームＳ２２の先頭のピクチャからデコード開始する時に必要なデコードバッファのビット占有量Ｓ２４が入力される。例えば、ビットストリームＳ２２が図２の(b)の第２のストリームＳＴ２である時、ビット占有量Ｓ２４は前記Ｏｓとなる。

【００７６】デコーダバッファシミュレータ１１２は、２つのビットストリームＳ２１とＳ２２を継ぎ合わせて伝送する時に、それがシームレスに画像再生できるように、これらビットストリームＳ１２とＳ２４に基づいて、前述同様にデコーダバッファへの入力ビットレートRinのスケジュールを立てる。すなわち例えば、前記図２の(c)のように、第２のストリームＳＴ２のｔ＝ｓｐ２におけるデコーダバッファのビット占有量がＯｓとなるように、直線ａ－ｋ－ｅ－ｆで示すように、デコーダバッファへの入力ビットレートを決める。より具体的に言うと、デコーダバッファシミュレータ１１２は、(1-1)式に基づいて、第１のストリームＳＴ１の最後ｔ＝ｓｐ１のビット占有量がＯｓになるように、ｔ＝αから入力ビットレートを変更するスケジュールを立てる。すなわち、デコーダバッファシミュレータ１１２は、図２の(c)の折れ線ａ－ｋ－ｅ－ｆで示されるデコーダバッファの入力ビットレートRin(t)のスケジュールを立てる。

【００７７】これにより、最終的にエンコーダバッファ１０３の出力ビットレートRoutは、デコーダバッファの入力ビットレートRinに同期して変化することになる。この制御方法は、上述の図３で説明したものと同じである。

【００７８】上述したようなことから、上記リアルタイムエンコードを行う場合であっても、デコーダバッファのオーバフロウやアンダフロウのような破綻を防ぐことが可能である。

【００７９】次ぎに図６には、図１のデコーダバッファシミュレータ１１２及び出力ビットレートコントローラ１０９によるエンコーダバッファ１０３の出力ビットレート制御の流れを示す。

【００８０】ステップｓｔ１では、Ｏｅ＝Ｏｓか否かを判断する。このステップｓｔ１にてｎｏと判断したときにはステップｓｔ３に進み、ｙｅｓと判断したときにはステップｓｔ２に進む。ステップｓｔ２では、第１のストリームＳＴ１のデコーダバッファへの入力ビットレートが変化しないように、エンコーダバッファの出力ビットレートを制御する。

【００８１】ステップｓｔ３では、Ｏｅ＜Ｏｓか否かを判断する。このステップｓｔ３にてｎｏと判断したときにはステップｓｔ５に進み、ｙｅｓと判断したときにはステップｓｔ４に進む。

【００８２】ステップｓｔ４では、Ｏｅ－Ｏｓとなるように第１のストリームＳＴ１のデコーダバッファへの入力ビットレートを時刻αから上げるように、エンコーダバッファの出力ビットレートを制御する。一方、ステップｓｔ５では、Ｏｅ＝Ｏｓとなるように第１のストリームＳＴ１のデコーダバッファへの入力ビットレートを時刻αから下げるように、エンコーダバッファの出力ビットレートを制御する。

【００８３】なお、本発明は、上述した実施の形態のみに限定されるものではなく、例えば取り扱うディジタル信号はビデオ信号に限定されず、オーディオ信号等にも適用できる。この他、本発明の要旨を逸脱しない範囲において、種々の変更が可能であることは勿論である。

【００８４】

【発明の効果】本発明は、ディジタル信号を符号化したビットストリームを伝送するための符号化信号伝送方法及び装置であり、入力ビットストリームを一時的に蓄える送信バッファを設け、短時間の入力符号量の変動を平滑化し、送信バッファから所定のビットレートでビットストリームを出力するシステムにおいて、２つのビットストリームを継ぎ合わせて伝送する時に、時間的に後ろとなるビットストリームをデコード開始する時に必要なデコーダ(受信)バッファの初期ビット占有量に基づいて、上記送信バッファからの出力ビットレートを制御しており、具体的には、２つのビットストリームを継ぎ合わせて伝送する時に、時間的に後ろとなるビットストリームをデコード開始する時のデコーダバッファのビット占有量が、そのストリームのデコードに必要な初期ビット占有量に等しいまたはそれ以上になるように、上記送信バッファからの出力ビットレートを制御することにより、２つのビットストリームを継ぎ合わせて伝送したとしても、デコーダ側でのシームレスな画像再生を保証できる。

【００８５】また、継ぎ合わせて伝送するところの２つのビットストリームは、既に符号化されているものでも良いので、汎用性が高い。また、リアルタイムエンコードするビットストリームを継ぎ合わせる場合にも、そのエンコード条件に制限を加える必要がないので、エンコーダシステムの構成を簡単化できる。

【図面の簡単な説明】

【図１】本発明の符号化信号伝送方法を実現する符号化信号伝送装置のブロック回路図である。

【図２】２つのビットストリームを継ぎ合わせて伝送する場合に、デコーダ側でデコーダバッファのアンダーフ

ロウを起こさないようにしてシームレスな再生を実現するための本発明の符号化信号伝送方法にかかるデコーダバッファのビット占有量の変化を示す図である。

【図３】図２の(c)に示したようにデコードバッファへの入力ビットレートを制御する時のエンコーダとデコーダのバッファのビット占有量の変化の例を示す図である。

【図４】２つのビットストリームを継ぎ合わせて伝送する場合に、デコーダ側でデコーダバッファのオーバフロウを起こさないようにしてシームレスな再生を実現するための本発明の符号化信号伝送方法にかかるデコーダバッファのビット占有量の変化を示す図である。

【図５】図４の(c)に示したようにデコードバッファへの入力ビットレートを制御する時のエンコーダとデコーダのバッファのビット占有量の変化の例を示す図である。

【図６】デコーダバッファシミュレータと出力ビットレートコントローラによるエンコーダバッファの出力ビットレート制御のフローチャートである。

【図７】符号化―多重化―分離―復号の一定遅延モデルを示す図である。

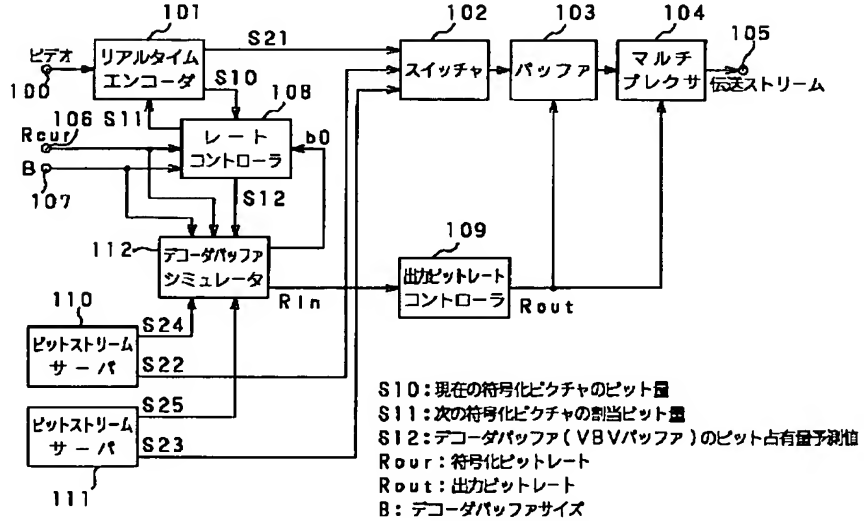
【図８】２つのビットストリームを継ぎ合わせて伝送した時に、デコーダバッファがアンダーフロウを起こしている例を示す図である。

【図９】２つのビットストリームを継ぎ合わせて伝送した時に、デコーバッファがオーバフロウを起こしている例を示す図である。

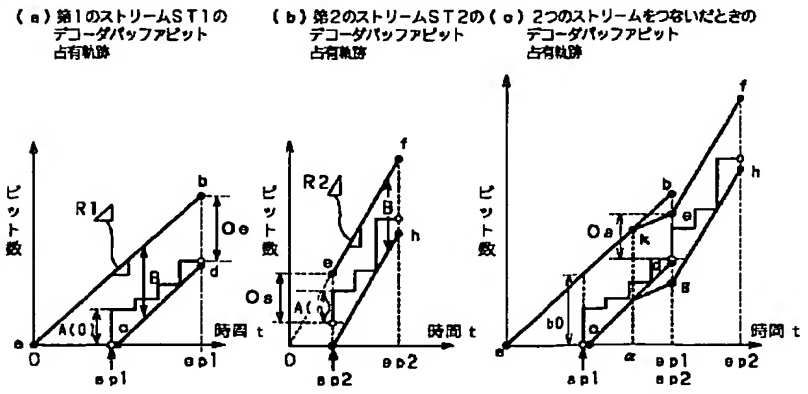
【符号の説明】

１０１　リアルタイムエンコーダ、　１０２　スイッチャ、　１０３　エンコーダバッファ、　１０４　マルチプレクサ、　１０８　レートコントローラ、１０９　出力ビットレートコントローラ、　１１０，１１１　ビットストリームサーバ、　１１２　デコーダバッファシミュレータ

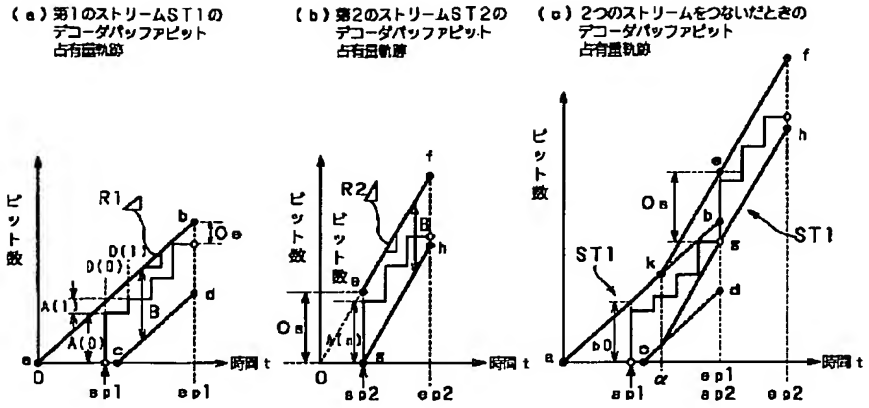
【図１】



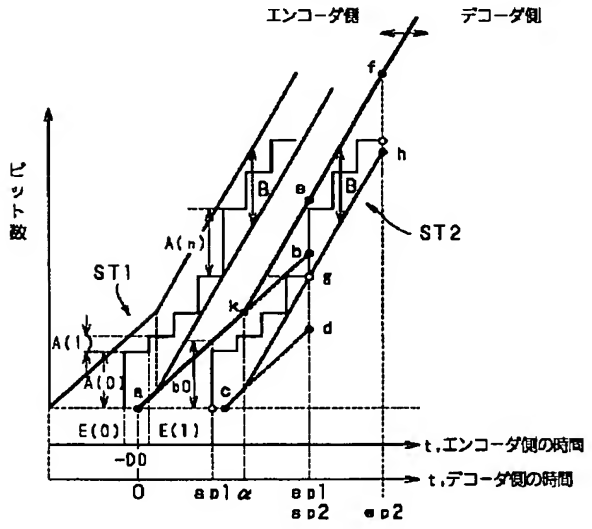
【図４】



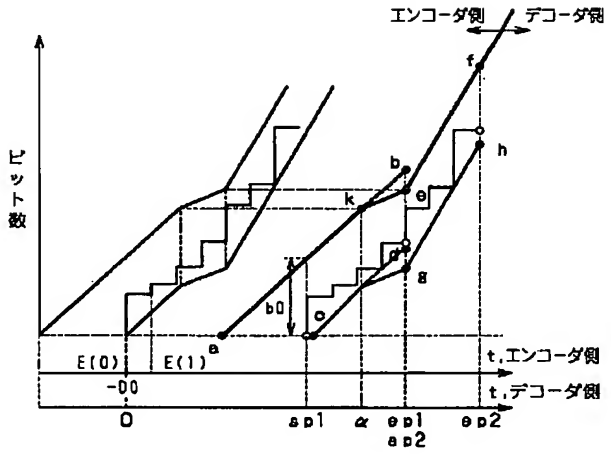
【図２】


（ａ）第1のストリームＳＴ1の
デコーダバッファビット
占有量軌跡
（ｂ）第2のストリームＳＴ2の
デコーダバッファビット
占有量軌跡
（ｃ）2つのストリームをつないだときの
デコーダバッファビット
占有量軌跡
ビット数
時間 t
R1
R2
ST1
ST2
A(n)
B
Oe
Os
asp1
asp2
esp1
esp2
α

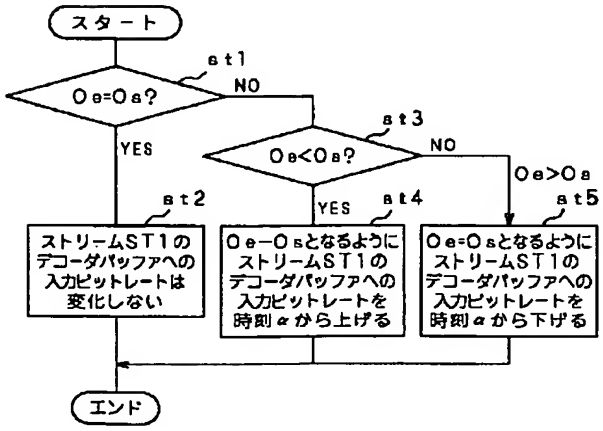
【図３】


エンコーダ側
デコーダ側
ビット数
ST1
ST2
A(n)
B
E(0)
E(1)
-DD
t,エンコーダ側の時間
t,デコーダ側の時間
asp1
esp1
asp2
esp2
α

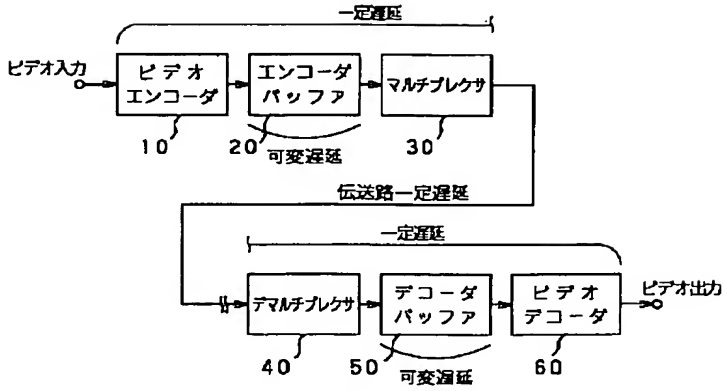
【図５】


エンコーダ側
デコーダ側
ビット数
E(0)
E(1)
-DD
t,エンコーダ側
t,デコーダ側
asp1
esp1
asp2
esp2
α

【図６】

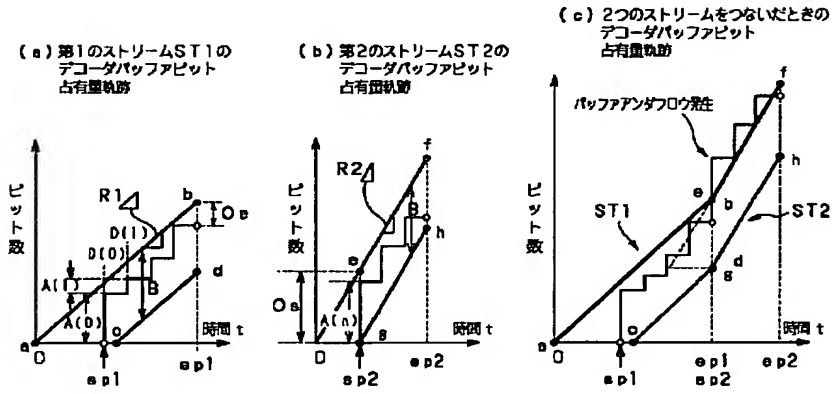

スタート
st1
Oe=Os?
NO
YES
st3
Oe<Os?
NO
YES
Oe>Os
st2
st4
st5
ストリームＳＴ1のデコーダバッファへの入力ビットレートは変化しない
Oe−Osとなるようにストリーム ＳＴ1のデコーダバッファへの入力ビットレートを時刻αから上げる
Oe=Osとなるようにストリーム ＳＴ1のデコーダバッファへの入力ビットレートを時刻αから下げる
エンド

【図７】



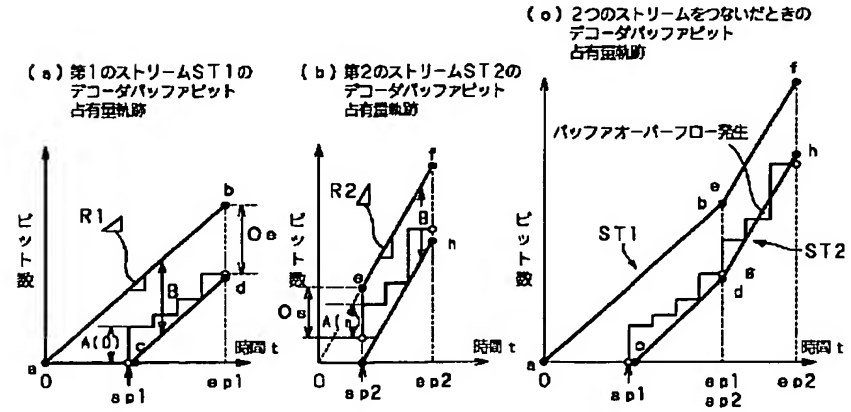
符号化・多重化・分離・復号の一定遅延モデル

【図８】



ビットストリームスプライシングによるデコーダバッファのアンダフロウ例

【図９】



ビットストリームスプライシングによるデコーダバッファのオーバフロウ例